CS 376 – Algorithm Design and Analysis

Professor Eric Aaron

Propositions

• Defn: *proposition* – a statement that has the property of truth or falsity
• Propositions are the key elements to represent, analyze, or explain declarative knowledge

Propositions:
- Washington, D.C. is the capital of the USA.
- Poughkeepsie is the capital of New York.
- \[ 1 + 1 = 2 \]
- \[ 2 + 2 = 3 \]

Non-Propositions:
- What time is it?
- Pass the salt.
- \[ x + 1 = 2 \]
- \[ x^y + 5z \] presuming values for \( x, y, z \) are not given / known

The first and third of these are true; the second and fourth are false.
Propositional operators

- Recall: proposition – a statement that has the property of truth or falsity
  - Often, we use propositional letters (or variables) to represent propositions: e.g., \( p \) stands for “Poughkeepsie is the capital of NY”

- There are several operators (sometimes called boolean operators) that can construct new propositions from old ones
  - Negation (“not”): if \( P \) is a proposition, \( \neg P \) is a proposition
  - Conjunction ("and"): \( P \) and \( Q \)
  - Disjunction ("or"): \( P \) or \( Q \)
  - Implication ("if – then"): if \( P \) then \( Q \)
  - Equivalence ("is equal / equivalent to"): \( P \iff Q \)
    - Equivalence can also be written as "if and only if"

Propositional operator: Negation

- Whatever the value of \( p \), True or False, the value of proposition \( \neg p \) (written \( \neg p \)) is the opposite
  - If \( p \) is “Today is Monday,” \( \neg p \) is “It is not the case that today is Monday,” or more simply “Today is not Monday.”
- Negation can be expressed with a truth table

<table>
<thead>
<tr>
<th>( p )</th>
<th>( \neg p )</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
</tr>
</tbody>
</table>
Propositional operator: Conjunction

- Conjunction—the “and” operator
  - Whatever the values of propositions $p$, $q$, conjunction $p$ and $q$ (written $p \land q$ or $p \& \& q$) is also a proposition
  - If $p$ is “Today is Monday” and $q$ is “It is snowing today,” then $p \land q$ is “Today is Monday and it is snowing today.”
    - $p \land q$ is true on snowy Mondays and false on any day that is not Monday, and on any day that is Monday but not snowing
- Conjunction values as a truth table

<table>
<thead>
<tr>
<th>$p$</th>
<th>$q$</th>
<th>$p \land q$</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
</tbody>
</table>

Propositional operator: Disjunction

- Disjunction—the “or” operator
  - Whatever the values of propositions $p$, $q$, disjunction $p$ or $q$ (written $p \lor q$ or $p \| q$) is also a proposition
  - If $p$ is “Today is Monday” and $q$ is “It is snowing today,” then $p \lor q$ is “Today is Monday or it is snowing today.”
    - $p \lor q$ is true on any day that is a Monday or on which it is snowing – including snowy Mondays (it is not exclusive) – and false only on days that are not Mondays on which it is not snowing
- Disjunction values as a truth table

<table>
<thead>
<tr>
<th>$p$</th>
<th>$q$</th>
<th>$p \lor q$</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
</tbody>
</table>

The non-exclusive sense of “or” can be a bit subtle

Exercise: What would the exclusive-or operator’s truth table look like?

It turns out there is such an operator, and it’s commonly used in logic! The English word “or” is a complicated thing to understand!
Propositional operator: Implication

- Implication—the “if…then” operator (also called conditional)
  - Whatever the values of propositions $p, q$, implication $p \implies q$ (written $p \rightarrow q$) is also a proposition
  - If $p$ is “Today is Monday” and $q$ is “It is snowing today,” then $p \rightarrow q$ is “If today is Monday then it is snowing today.”
  - Vocabulary: in $p \rightarrow q$, $p$ is called the hypothesis (or antecedent) and $q$ is called the conclusion (or consequent)

- Implication values as a truth table

<table>
<thead>
<tr>
<th>$p$</th>
<th>$q$</th>
<th>$p \rightarrow q$</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
</tr>
</tbody>
</table>

Really? These are the truth values for implication? They look like the values for ($\neg p \lor q$)! (Exercise: Check for yourselves!!)

Sounds if-y: Material Implication

- Meaning for implication symbol $\rightarrow$ in propositional logic is referred to as material implication
  - It says that $p \rightarrow q$ is False exactly when $p$ is True and $q$ is False
  - Not the same as every meaning of “if…then” in English, but it’s what’s used in logic

Examples of material implication and natural language usage:
- Politician says: “If I am elected, then I will fix the environment”
  - False if the speaker is elected and doesn’t fix the environment
  - True if, e.g., the speaker doesn’t get elected
- “If today is Friday, then $2 + 2 = 4$”
  - True no matter what day it is
- “If today is Friday, then $2 + 2 = 5$”
  - True except on Fridays, even though $2 + 2 = 5$ is false!
Properties of operators

• Logical operators have an order of operations just like mathematical operators
  – From high to low: negation; conjunction; disjunction; implication
  • Conjunction is kinda like multiplication; disjunction is kinda like addition
    • Math: -k * (x + y)
    • Logic: \( \neg p \land (q \lor r) \)
• Also similarly, disjunction and conjunction are commutative and associative
  – Associative: e.g., \( p \land q \land r \) is \( (p \land q) \land r \)
  – Commutative: e.g., \( p \land q \) is \( q \land p \)
  • similar with disjunction
• Implication is right-associative
  – \( p \rightarrow q \rightarrow r \) is \( p \rightarrow (q \rightarrow r) \)

The biconditional (or equivalence) operator

• The biconditional (or equivalence) operator:
  – If \( p \) and \( q \) are propositions, then \( p \leftrightarrow q \) is a proposition, read as “\( p \) if and only if \( q \)”
  – \( p \leftrightarrow q \) is true exactly when \( p \) and \( q \) have the same truth values
• What does the truth table for \( \leftrightarrow \) look like?
• How could we define the biconditional in terms of operators we already know (not, and, or, if… then)?

The equivalence operator can also be written as \( = \) or \( == \) in other contexts.
Exercise: Evaluating boolean expressions

- Defn: Propositions are boolean-valued expressions—i.e., their values are either True or False
- Boolean expressions are evaluated like any other mathematical expressions

Examples: Let p = True, q = False, r = True. What do the following expressions evaluate to?
1. \( p \land \neg r \)
2. \( q \lor False \)
3. \( p \rightarrow q \)
4. \( q \leftrightarrow p \)
5. \( q \leftrightarrow \neg True \)
6. \( r \lor (p \land q) \)
7. \( (p \lor r) \rightarrow ((p \lor q) \land r) \)
8. \( True \rightarrow r \)
Compound propositions and their truth tables

- Just as we use truth tables to understand meanings of propositional operators, we can also use them to understand compound propositions.
  - The truth table for \((p \lor \neg q) \rightarrow (p \land q)\):

<table>
<thead>
<tr>
<th></th>
<th>(p)</th>
<th>(q)</th>
<th>(\neg q)</th>
<th>(p \lor \neg q)</th>
<th>(p \land q)</th>
<th>((p \lor \neg q) \rightarrow (p \land q))</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
</tbody>
</table>
Exercise: Truth tables for compound propositions

- What are truth tables for the following propositions?
  1. $p \rightarrow \neg p$
  2. $p \leftrightarrow \neg p$
  3. $(p \rightarrow q) \land (\neg p \rightarrow q)$
  4. $(p \lor q) \land r$
  5. $p \rightarrow (\neg q \lor r)$